VIT-AP UNIVERSITY

ECS1002

Weather Monitoring System

Members Involved:	Sabareenath S Pillai	– 21BCE8865
	Bharath Nair	– 21BCE8248
	Yedu Krishnan Satyan	– 21BCE8367
	Habel Subash Pallicken	– 21BCE8741
	Sidharth Gopalakrishnan	– 21BCE8243
	Cheethirala Vamsi Vivek	– 21BCE7266,

Under the guidance of Prof. Sucharitha Jackson

Summary of the Project

This project helps create a weather monitoring system using the Blynk app. This project is mainly based on three sensors. That is the rain sensor, DHT11 sensor and LDR sensor. Through this, we can see factors such as rainfall, temperature, humidity, and amount of light. Also, the speciality is that we can monitor all this over the internet. Since this is accessible over the internet it can be monitored anywhere at any time. The technology behind this is the Internet of Things (IoT), which is an advanced and efficient solution for connecting things to the internet and connecting the entire world of things in a network. Here things might be whatever like electronic gadgets, sensors, and automotive electronic equipment. The data updated from the implemented system can be accessible in the internet from anywhere in the world. Also, we can do this project at a low cost and use it for farms and greenhouses.

INDEX

S No.	Title
1	LIST OF FIGURES
2	LIST OF ABBREVATIONS
3	INTRODUCTION
4	OBJECTIVES
5	PROCEDURE
6	RESULT
7	CONCLUSION & FUTURE SCOPE
8	REFERENCES
9	CODES

LIST OF FIGURES

S No.	Component Name
1	NodeMCU ESP8266 Board
2	Rain sensor
3	DHT11 sensor
4	LDR sensor
5	Jumper wires
6	Block Diagram

LIST OF ABBREVATIONS

1) IoT – Internet of Things
2) DHT- Digital Humidity and Temperature
3) LDR- Light Dependent Resistor
4) ESP- Espressif Systems.
5) LED- Light Emitting Diode

INTRODUCTION

Here we introduce a smart weather reporting system over the Internet. Our introduced system allows for weather parameter reporting over the Internet. It allows the people to directly check the weather status online without the need of a weather forecasting agency. System uses temperature, humidity as well as rain with humidity sensor to monitor weather and provide live reporting of the weather statistics. The system constantly monitors temperature using temperature sensor, humidity using humidity sensor and also for rain. Weather monitoring system deals with detecting and gathering various weather parameters at different locations which can be analysed or used for weather forecasting. The aim of this system is achieved by technologies such as Internet of Things (IOT) and Cloud. The idea of internet of things is to connect a device to the internet and to other required connected devices. Using internet, the information from the IOT device can easily be transferred to the cloud and then from the cloud to the end user. Weather Monitoring is an essential practical implementation of the concept of Internet of Things, it involves sensing and recording various weather parameters and using them for alerts, sending notifications, adjusting appliances accordingly and also for long term analysis.

OBJECTIVES

The objectives of this project are to:

1. Design the circuit using Nodemcu with some modern reliable sensors and other components.
2. Write the code using the C++ computer programming language.
3. Use the system to log in weather data to ensure that the device measures weather data periodically.
4. Approximate the dew point temperature and humidity.

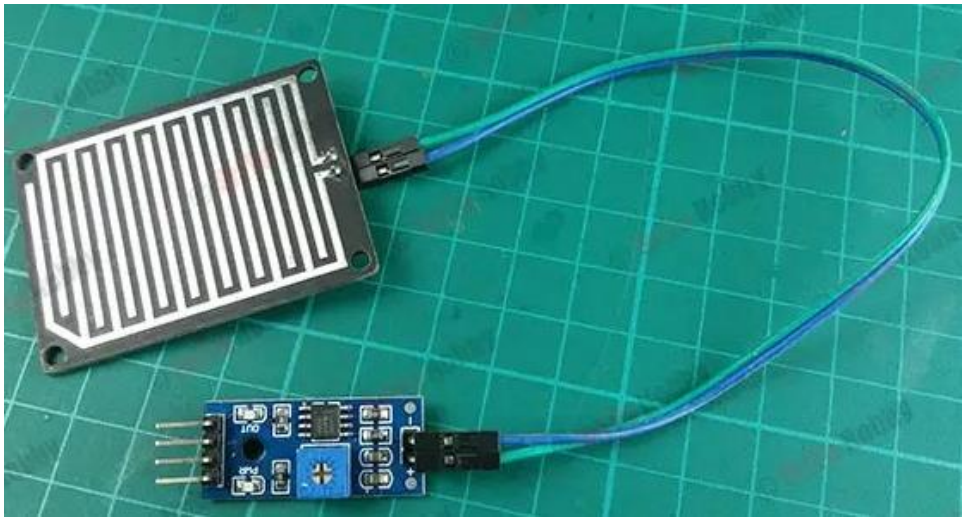
PROCEDURE

STEP 1 - Firstly, identify these components.

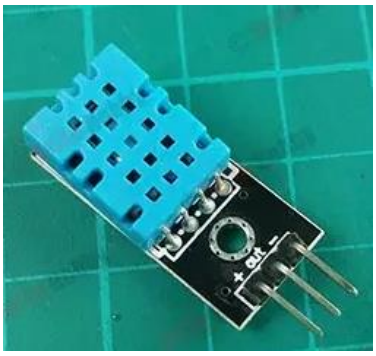
The required components are given below.

1) NodeMCU ESP8266 board - is a self-contained SOC with in an integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor.

2) Rain sensor - A rain sensor is one kind of switching device which is used to detect the rainfall. It works like a switch and the working principle of this sensor is, whenever there is rain, the switch will be normally closed. Basically, this board includes nickel coated lines and it works on the resistance principle. This sensor module permits to gauge moisture through analog output pins & it gives a digital output while moisture threshold surpasses.



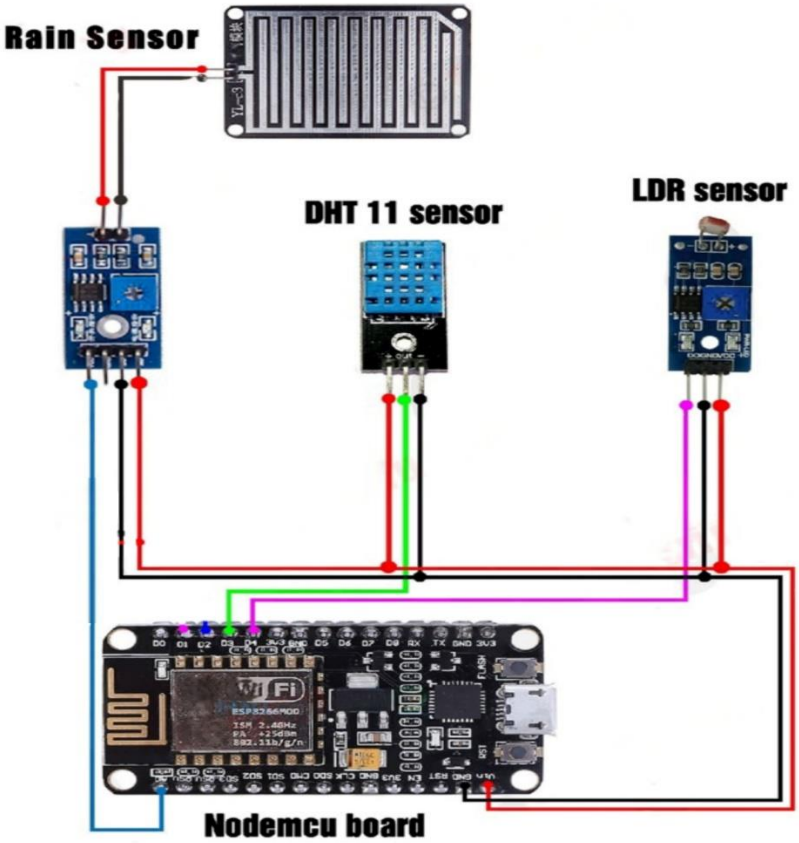
3) DHT11 sensor - is a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analog input pins needed)



4) LDR sensor - tiny light-sensing devices also known as photoresistors. An LDR is a resistor whose resistance changes as the amount of light falling on it changes. The resistance of the LDR decreases with an increase in light intensity

- Jumper wires

STEP 2 - Secondly, connect these components. To do this, use the circuit diagram below.

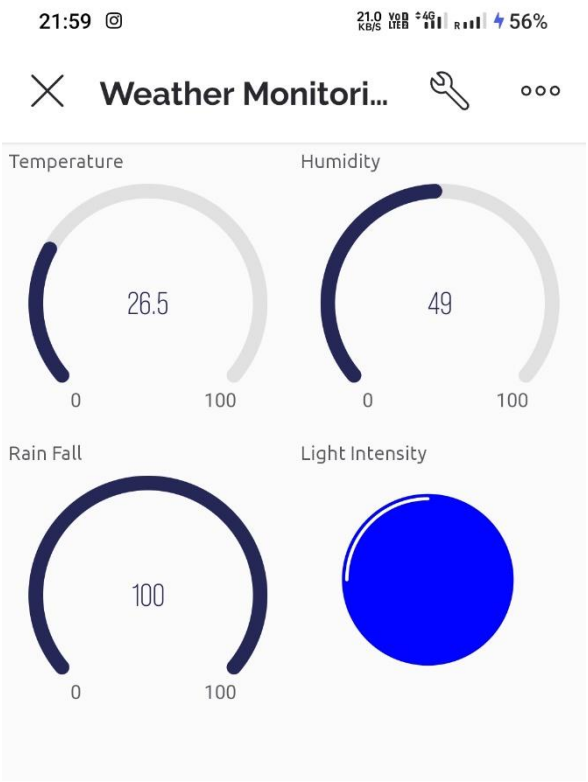


STEP 3 - Thirdly, let's set up the Blynk app. For that, follows the steps below.

- First, downloading and install the Blynk app on your phone. After, sign up for this app using our email address. Then, click the "New project" button.
- Next, enter the project name as you like and select the device and connection type. Then click the "Confirm" button.
- OK, now we can see the project interface. Next, let's add the widgets to the interface. For that, click the "+" icon in the corner and includes the three Gauge widgets and one LED widget.
- OK, let's set up this widget one by one. For that, Name the Gauge widgets as Temperature, Humidity and Rainfall, respectively. Also, set the PINs to V0, V1 and V2 and set the display values 0 to 100.
- Then, clicking the LED widget and set the PIN as V3.

STEP 4 - Create the program for this weather monitoring system. Select board and port. Afterward, upload this code to the NodeMCU board.

RESULT


21:59
56%
Weather Monitori...
Temperature
26.5
0
100
Humidity
49
0
100
Rain Fall
100
0
100
Light Intensity

CONCLUSION & FUTURE SCOPE

The technology of IoT has expanded in all sectors, and with the future scope and advantages of IoT-based weather monitoring systems, numerous industries can leverage them.

The IoT weather reporting system has an application for farmers where they can ensure higher productivity of crops and lower the risk of weather hazards via the IoT weather.
The IoT-based weather station proves helpful for monitoring the weather in areas like places with volcanoes or rain forests. This is especially important with drastic changes in the weather conditions we are experiencing.
The IoT weather monitoring system using IoT supporting controllers is fully automated and efficient. It does not require any manual labour or attention.

We can plan and visit the places anytime you like with prior notification of the weather conditions. We can simply get the status of the weather condition and the air quality, etc.
Therefore, with the help of embedded devices and sensors, any environment can be converted to a smart environment for accumulating the data and analysing the environment with real-time monitoring.

Hence, with such advances on the Internet of Things (IoT), organizations are focusing on understanding the impact of weather on their operations and finding cutting-edge analytics on how to control the impact of their business.

REFERENCES

1) https://www.airtel.in/blog/business/iot-use-cases-in-real-time-weather-monitoring-system/#:~:text=The%20IoT%20weather%20reporting%20system,with%20volcanoes%20or%20rain%20forests.

2) https://srituhobby.com/iot-based-weather-monitoring-system-using-nodemcu-and-blynk/

3) https://en.wikipedia.org/wiki/ESP8266

4) https://diyi0t.com/rain-sensor-tutorial-for-arduino-and-esp8266/

CODES

```
#include <LiquidCrystal_I2C.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
DHT dht(D3, DHT11); //(sensor pin,sensor type)
BlynkTimer timer;

char auth[] = ""; //Enter the Auth code which was send by Blink
char ssid[] = "Blynk";  //Enter your WIFI Name
char pass[] = "12345678";  //Enter your WIFI Password

void weather() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  int r = analogRead(A0);
  bool l = digitalRead(D4);

  r = map(r, 0, 1023, 100, 0);
  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Blynk.virtualWrite(V0, t);  //V0 is for Temperature
  Blynk.virtualWrite(V1, h);  //V1 is for Humidity
  Blynk.virtualWrite(V2, r);  //V2 is for Rainfall

  if (l == 0) {
    WidgetLED led1(V3);
    led1.on();
    lcd.setCursor(9, 1);
    lcd.print("L :");
    lcd.print("High");
    lcd.print(" ");
  } else if (l == 1) {
    WidgetLED led1(V3);
    led1.off();
    lcd.setCursor(9, 1);
    lcd.print("L :");
    lcd.print("Low");
    lcd.print(" ");
  }
```

```
  lcd.setCursor(0, 0);
  lcd.print("T :");
  lcd.print(t);

  lcd.setCursor(0, 1);
  lcd.print("H :");
  lcd.print(h);

  lcd.setCursor(9, 0);
  lcd.print("R :");
  lcd.print(r);
  lcd.print("  ");

}

void setup() {
  Serial.begin(9600);  lcd.init();
  lcd.backlight();
  Blynk.begin(auth, ssid, pass);
  dht.begin();
  timer.setInterval(10L, weather);
}

void loop() {
  Blynk.run(); // Initiates Blynk
  timer.run(); // Initiates SimpleTimer
}
```